## Lab 4 Specifications

### Lab-specific Specifications

### Proficiency

- ☐ Design plays *Für Elise* from provided starter code
- ☐ Note durations match the durations specified in the starter code for *Für Elise* (i.e., the tune plays at the correct tempo)
- ☐ Individual pitches are accurate to within 1% across the frequency range (of 220-1000 Hz) (calculations should be provided in the report to verify this)
- ☐ All rests (pauses with no sound) are played properly
- ☐ Code uses `#define` macros for memory-mapped registers
- ☐ Portfolio page includes a video of the system playing the entire song.

### Excellence

- ☐ Report contains accurate calculations for **minimum duration** supported
- ☐ Report contains accurate calculations for **maximum duration** supported
- ☐ Report contains accurate calculations for **minimum frequency** supported
- ☐ Report contains accurate calculations for **maximum frequency** supported
- ☐ Report provides documentation and calculations to show that the durations and pitches are correct based on the timer configuration.
- ☐ Design contains potentiometer to control the output volume.
- ☐ Design plays an extra composition of your choice. You need not compose the tune from scratch, it is acceptable to transpose an existing tune.

### General Specifications

### Proficiency

### General Schematic Specifications

- ☐ All pin names labeled
- ☐ All pin numbers labeled
- ☐ Crossing wires clearly identified as junction or unconnected
- ☐ Neat layout (e.g., clear organization and spacing)
- ☐ All parts labeled with part number
- ☐ All component values present

### Block Diagram

- ☐ Block diagram present with one block per SystemVerilog module
- ☐ Each block includes all input and output signals

### HDL & Code Specifications

*General Formatting*

- ☐ Descriptive filename (e.g., `lab2_jb.sv`)
- ☐ Descriptive variable names
- ☐ Neat formatting (e.g., standard indentation, consistent formatting for variable names (kebab-case/snake_case/camelCase/PascalCase ))
- ☐ Descriptive and clear function/module names

*Comments*

- ☐ Comments to indicate the purpose of each function/module

### Lab Writeup/Summary

- ☐ Brief (e.g., 3-5 sentence) description of the main goals of the assignment and what was done.
- ☐ Explanation of design approach. How did you go about designing and implementing the design?
- ☐ Explanation of testing approach. How did you verify your design was behaving as expected?
- ☐ Statement of whether the design meets all the requirements. If not, list the shortcomings.
- ☐ Number of hours spent working on the lab are included.
- ☐ Writeup contains minimal spelling or grammar issues and any errors do not significantly detract from clarity of the writeup.
- ☐ (Optional) List comments or suggestions on what was particularly good about the assignment or what you think needs to change in future versions.

## Excellence

### General Schematic Specifications

- ☐ Standard symbols used for all components where applicable
- ☐ Signals "flow" from left to right where possible (e.g., inputs on left hand side, outputs on right hand side)
- ☐ Title block with author name, title, and date

### HDL & Code Specifications

*General Formatting*

- ☐ Name, email, and date at the top of every file
- ☐ Comment at the top of each source code file to describe what is in it
- ☐ Clear and organized hierarchy (e.g., delineation between top level modules and submodules)

*Testbenches*

- ☐ Testbenches written for each individual module to demonstrate proper operation
- ☐ Testbench output included in the report

### Lab Writeup/Summary

- ☐ Writeup is free of spelling and grammar issues

## Comments

Add specific notes here about the assignment.